# NAVAL POSTGRADUATE SCHOOL
## Monterey, California


# THESIS

**AN ANALYSIS OF QUALITY OF SERVICE OVER THE AUTOMATED DIGITAL NETWORK SYSTEM**

by

Brian D. Rehard

September, 1997

Thesis Advisor: Rex Buddenberg

# REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188



| 1. AGENCY USE ONLY *(Leave blank)* | 2. REPORT DATE<br>September 1997 | 3. REPORT TYPE AND DATES COVERED<br>Master's Thesis |
|---|---|---|

| 4. TITLE AND SUBTITLE An Analysis of Quality of Service Over the Automated Digital Network System | 5. FUNDING NUMBERS |
|---|---|
| 6. AUTHOR(S) Brian D. Rehard | |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>Naval Postgraduate School<br>Monterey CA 93943-5000 | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSORING/MONITORING AGENCY REPORT NUMBER |

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

| 12a. DISTRIBUTION/AVAILABILITY STATEMENT<br>Approved for public release; distribution is unlimited. | 12b. DISTRIBUTION CODE |
|---|---|

13. ABSTRACT *(maximum 200 words)*

With the implementation of the Automated Digital Network System (ADNS), the United States Navy has significantly expanded its communication capabilities. However, as ADNS is installed throughout the Fleet, and bandwidth-hungry applications such as video teleconferencing become more popular, network congestion will become a larger and larger problem. Specifically, network congestion will cause a slow down in delivery of all traffic. Applications with hard, real-time requirements for data delivery, which treat late message packets as lost packets, will begin to lose data. This thesis will explore the message priority setting and congestion handling functions of ADNS, pointing out inadequacies during congested conditions which may lead to data losses. It will then go on to introduce Quality of Service (QoS) standards being developed by the Internet Engineering Task Force (IETF). These QoS standards are implemented by reservation protocols to provide deterministic service over networks regardless of network loading. Finally this thesis will introduce the Resource Reservation Protocol (RSVP) as a means to implement QoS over ADNS, allowing privileged applications to enjoy deterministic service over the network at any time or under any conditions of network loading.

| 14 SUBJECT TERMS *Type the keywords (at least three) for your thesis in over these words; all keywords must be unclassified. | 15. NUMBER OF PAGES 88 |
|---|---|
| | 16. PRICE CODE |

| 17. SECURITY CLASSIFICATION OF REPORT<br>Unclassified | 18. SECURITY CLASSIFICATION OF THIS PAGE<br>Unclassified | 19. SECURITY CLASSIFICATION OF ABSTRACT<br>Unclassified | 20. LIMITATION OF ABSTRACT<br>UL |
|---|---|---|---|

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

**Approved for public release; distribution is unlimited.**

# AN ANALYSIS OF QUALITY OF SERVICE OVER THE AUTOMATED DIGITAL NETWORK SYSTEM

Brian D. Rehard
Lieutenant, United States Navy
B.S., Ohio State University, 1989

Submitted in partial fulfillment
of the requirements for the degree of

**MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT**

from the

**NAVAL POSTGRADUATE SCHOOL**
**September 1997**

## ABSTRACT

With the implementation of the Automated Digital Network System (ADNS), the United States Navy has significantly expanded its communication capabilities. However, as ADNS is installed throughout the Fleet, and bandwidth-hungry applications such as video teleconferencing become more popular, network congestion will become a larger and larger problem. Specifically, network congestion will cause a slow down in delivery of all traffic. Applications with hard, real-time requirements for data delivery, which treat late message packets as lost packets, will begin to lose data. This thesis will explore the message priority setting and congestion handling functions of ADNS, pointing out inadequacies during congested conditions which may lead to data losses. It will then go on to introduce Quality of Service (QoS) standards being developed by the Internet Engineering Task Force (IETF). These QoS standards are implemented by reservation protocols to provide deterministic service over networks regardless of network loading. Finally this thesis will introduce the Resource Reservation Protocol (RSVP) as a means to implement QoS over ADNS, allowing privileged applications to enjoy deterministic service over the network at any time or under any conditions of network loading.

## TABLE OF CONTENTS

# I. INTRODUCTION

## A. BACKGROUND

This thesis presents research conducted in the areas of wireless communication networks and network congestion handling. The U.S. Navy has recently installed the Automated Digital Network System (ADNS) aboard select Pacific Fleet ships for testing. This system integrates existing communication channels with Local Area Networks (LANs) aboard ships to form a Radio Wide Area Network (Radio-WAN) consisting of several ships and shore stations. ADNS enables Internet-like communications between ships and shore stations in a manner that is totally transparent to the user. Like the Internet, this Radio-WAN will be subject to congestion which can result in degradation of service. Providing a means to ensure vital communications receive quality service without degradation during times of network congestion will be integral to the success of ADNS.

## B. OBJECTIVES

The initial objective of this thesis is to provide the reader with an introduction to ADNS, specifically pointing out the priority setting and congestion handling features that ADNS offers, and the shortcomings of each. Secondly, this thesis provides a brief discussion of Naval communication requirements and capabilities. Third, an introduction to Quality of Service (QoS) over networks is presented. Finally, the Resource Reservation Protocol (RSVP) is proposed as a method of providing QoS over ADNS.

## C. METHODOLOGY

The subjects of this research; ADNS, QoS, and RSVP, are all in their infancy. As such, published works on these topics do not exist. The information presented in this thesis was gleaned from working papers, Internet-Drafts, and websites, as well as personal interviews with and training classes taught by the engineers who devised ADNS. Information on the history of Naval communication capabilities and requirements was obtained from historic documents as well as military websites.

## D. ORGANIZATION OF STUDY

The information presented in this thesis is divided into six chapters. Following this introduction included in Chapter I, Chapter II provides the reader with an introduction to ADNS. This description of ADNS is presented from an Information Technology Manager's point of view. This is a significant departure from any of the existing works on ADNS, most of which are technical documents written from an engineer's point of view. This chapter was written by the author in conjunction with LCDR Jim Sullivan and LT Eric Andalis, and is intended to be submitted as an informational Request for Comments (RFC) to the Internet Engineering Task Force (IETF).

Chapter III provides background information on the history of Naval communications and communications requirements vs. current capabilities. It then goes on to define and show a pressing need for deterministic service over ADNS. This chapter is the author's original work supported by facts and figures taken from military websites.

Chapter IV introduces QoS, providing IETF definitions of QoS standards. This information is largely taken from Internet-Drafts on QoS and, as such, is only preliminary in nature.

Chapter V introduces the Resource Reservation Protocol (RSVP), a means of implementing QoS over a terrestrial network. Again, the information presented in this chapter is largely taken from Internet-Drafts which are preliminary in nature and subject to revision.

Finally, Chapter VI presents analysis, conclusions and questions for further study. This represents the authors own work in identifying a possible service problem which may arise in ADNS, proposing the implementation of IETF QoS standards to solve this problem, and further proposing RSVP as the vehicle to implement QoS standards over ADNS.

# II. AUTOMATED DIGITAL NETWORK SYSTEM

ADNS is designed to implement a WAN over Radio Frequency (RF) links between mobile platforms. This WAN transports Internet Protocol (IP) datagrams over legacy RF links in a manner totally transparent to the applications which are sending and receiving information. A high-level context diagram of ADNS is presented in Figure 2.1.

ADNS provides the interface between existing Local Area Networks (LANs) and the RF links. These RF links may be direct High Frequency (HF) communications, or commercial or military satellite communications. ADNS does not provide any additional transmission channels, although it does provide for expansion. New channels may be added in the future. Instead, ADNS uses existing channels to form an Internet-like WAN which has far greater capabilities and flexibility than the traditional stovepipe communications architecture. Thus, ADNS improves communications not by adding more communication channels, but by implementing a Radio-WAN over existing channels (which includes built-in expandability to accomodate new channels in the future.)

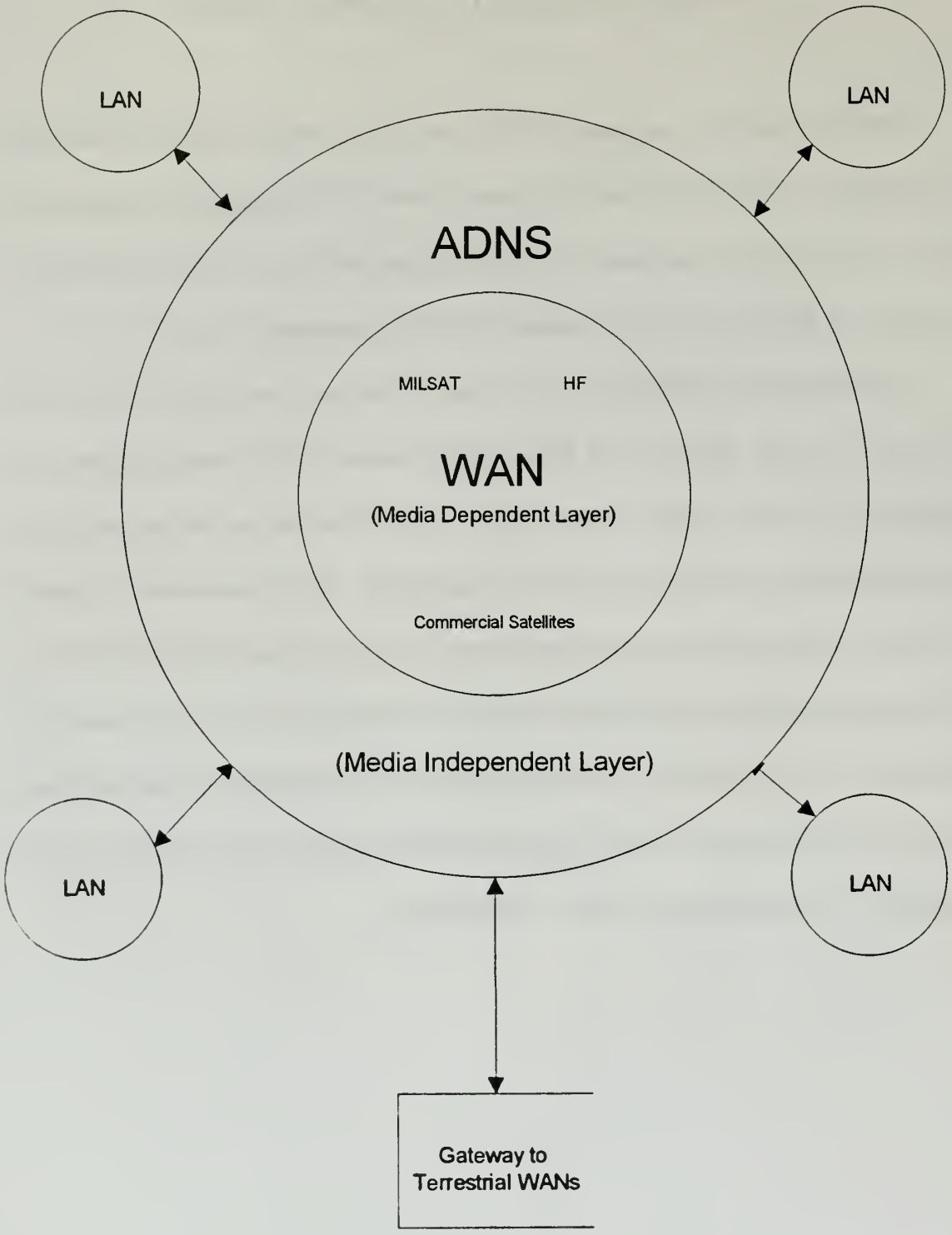


Figure 2.1 ADNS Context Diagram

## A. ADNS OVERVIEW

In order to fully explain the operation of ADNS, the components of ADNS must first be described. Figure 2.2 represents the inner workings of the ADNS "black box" depicted in Figure 2.1.

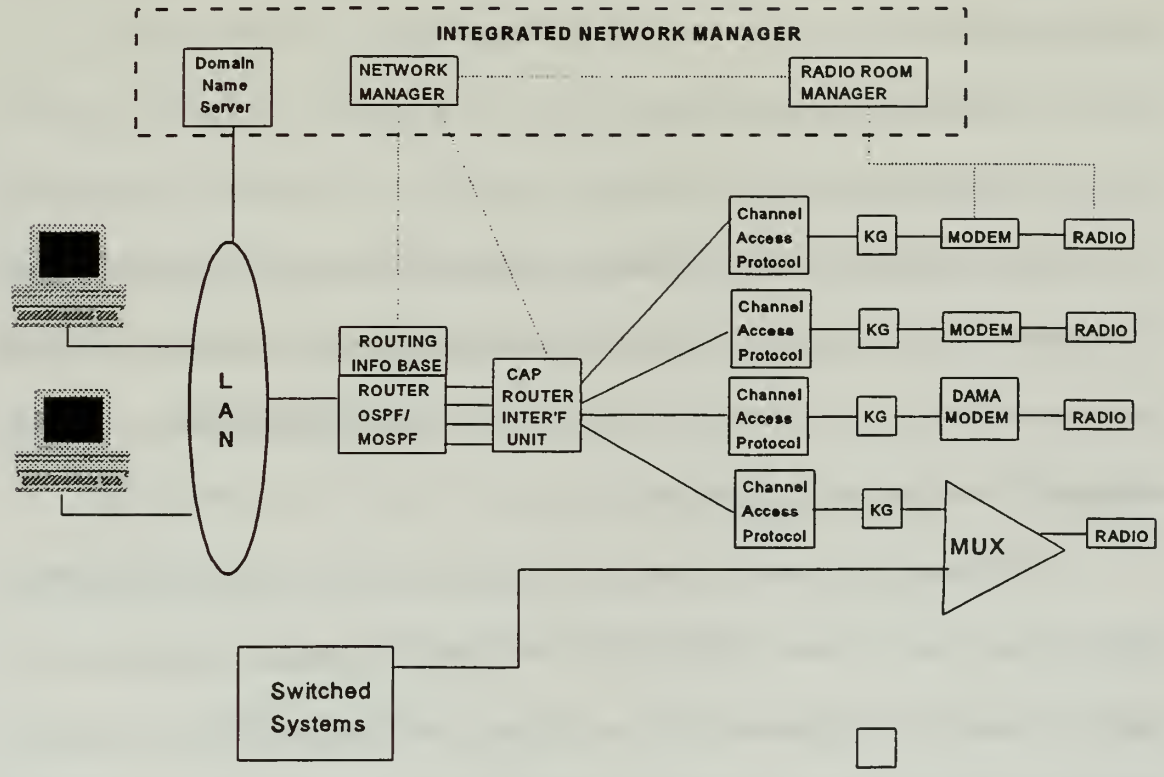


Figure 2.2 ADNS Configuration (Casey, 1997)

A typical message transmission through ADNS proceeds as follows. The router accepts outbound datagrams from the LAN and selects the best path for reaching the destination. The CAP Router Interface Unit (CRIU), which interfaces between the router and

Channel Access Protocol (CAP), assigns a priority to outbound IP datagrams. At the CAP the message is placed in a queue to await transmission. Messages in the CAP queue are sorted by the priority assigned by the CRIU. When the message leaves the CAP it passes through a cryptographic device. The standard Navy ADNS configuration operates at the secret high level of classification, thus all information entering the RF network is link encrypted. This:

1. Conforms to existing practice.

2. Provides resistance to AS spoofing.

3. Provides limited content confidentiality/authenticity protection (because this layer of encryption is stripped off at each routing point). Although this provides protection during transmission it does not provide content security once the information passes through the cryptographic device at the receiving end.

4. Provides opportunities for secure tunnels such as Unix Secure Shell (SSH) or Network Encryption System (NES), which deal with IP datagram encapsulation (IP datagrams inside other datagrams). These encapsulated IP datagrams are transmitted by ADNS in the same manner as any other IP datagrams.

5. Does not affect applications that offer end-to-end security (e.g. secure e-mail). Similar to secure tunnels, end system encrypted datagrams are unaffected by the presence of ADNS in the system.

After leaving the Cryptographic device the datagram passes through a modem and then enters the transmitter. Upon leaving the transmitter, the RF signal travels node to node

along the predetermined path to its destination. Upon arrival at its destination, the datagram passes through a mirror image of the originating system and terminates at the host specified in the IP header.

## B. KEY FEATURES OF ADNS

### 1. Logical Organization

ADNS nodes are logically organized into Autonomous Systems (AS). An AS consists of a number of nodes related by function, not necessarily geography. A carrier battlegroup is an example of an AS related by function. The battlegroup has a common purpose and need for connectivity, but the ships in the battlegroup may be separated by hundreds or even thousands of miles during the course of a deployment.

### 2. Routing Protocols

ADNS uses three different routing protocols. The primary reason for using these algorithms was that the specifications for all three are in the public domain.

#### *a. Open Shortest Path First (OSPF)*

OSPF is used as the Internal Gateway Protocol (IGP) for routing within an AS. It is a dynamic protocol in that each router maintains a continuously updated database containing the status of all other routers in the same AS. OSPF uses a lowest cost algorithm to determine the best path to send a message to its destination. Costs are determined based on metric values assigned to the various transmission paths. These metric values are currently assigned based on link capacity, with high capacity links having lower metric values indicating lower costs for transmission.

### *b. Multicast OSPF (MOSPF)*

Multicast OSPF (MOSPF) is used for multicast sessions within an AS. MOSPF uses the same lowest cost concept as OSPF with the exception that the lowest cost path is determined with respect to the entire multicast group instead of a single path.

### *c. Border Gateway Protocol Version 4 (BGP4)*

BGP4 is used as the External Gateway Protocol (EGP) for routing between ASs. BGP4 is not as dynamic as OSPF and makes its routing decisions based on predetermined routes. In ADNS, BGP4 will typically reside at the shore station in a system. Since BGP4 requires a more stable environment than OSPF, the shore station is the logical choice. The AS boundaries and routing protocols are illustrated in Figure 2.3.

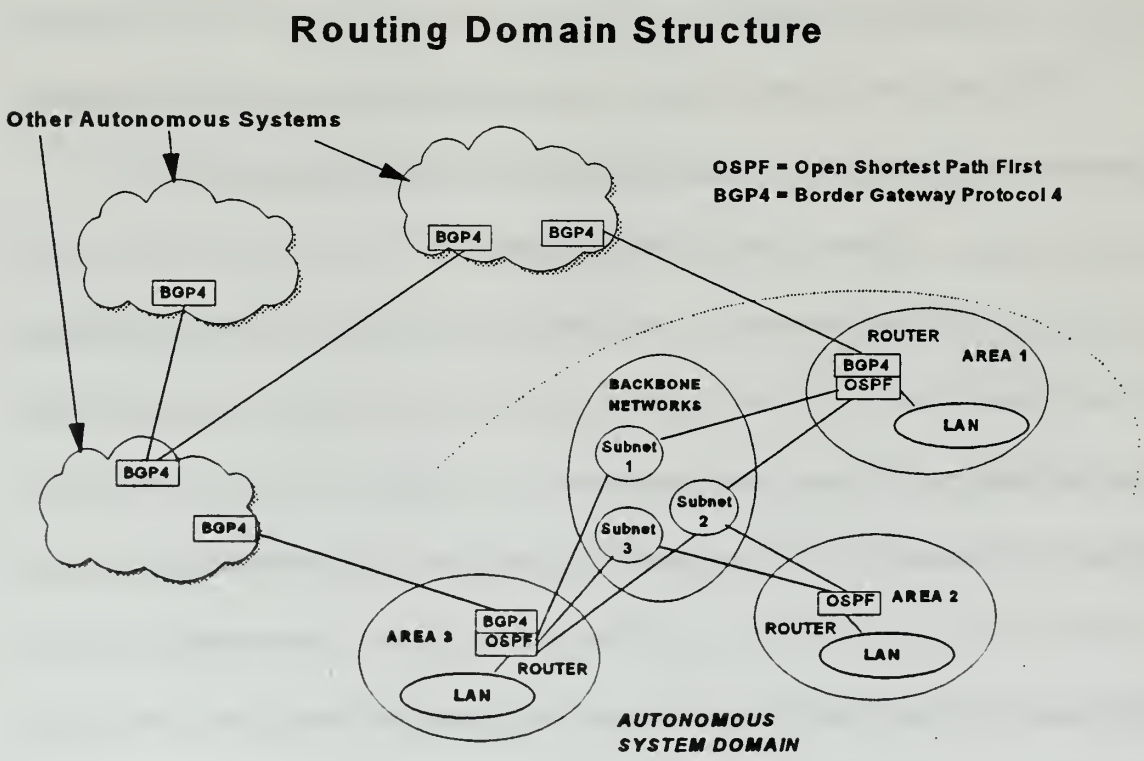


Figure 2.3 Autonomous System Boundaries and Routing Protocols (Casey, 1997)

### 3. Priority

Several different methods for assigning priority to outgoing messages were evaluated during the ADNS implementation process. One obvious method, using the built-in precedence field in the IP header, was briefly considered. This idea was quickly discarded since no relevant applications currently use this feature of the IP header. Eventually, a priority scheme was implemented which assigned priorities of 0 (lowest) to 15 (highest). The two methods which proved most useful for assigning priority were based on source IP address (Host), or port number (Application). This process is illustrated in Figure 2.4.

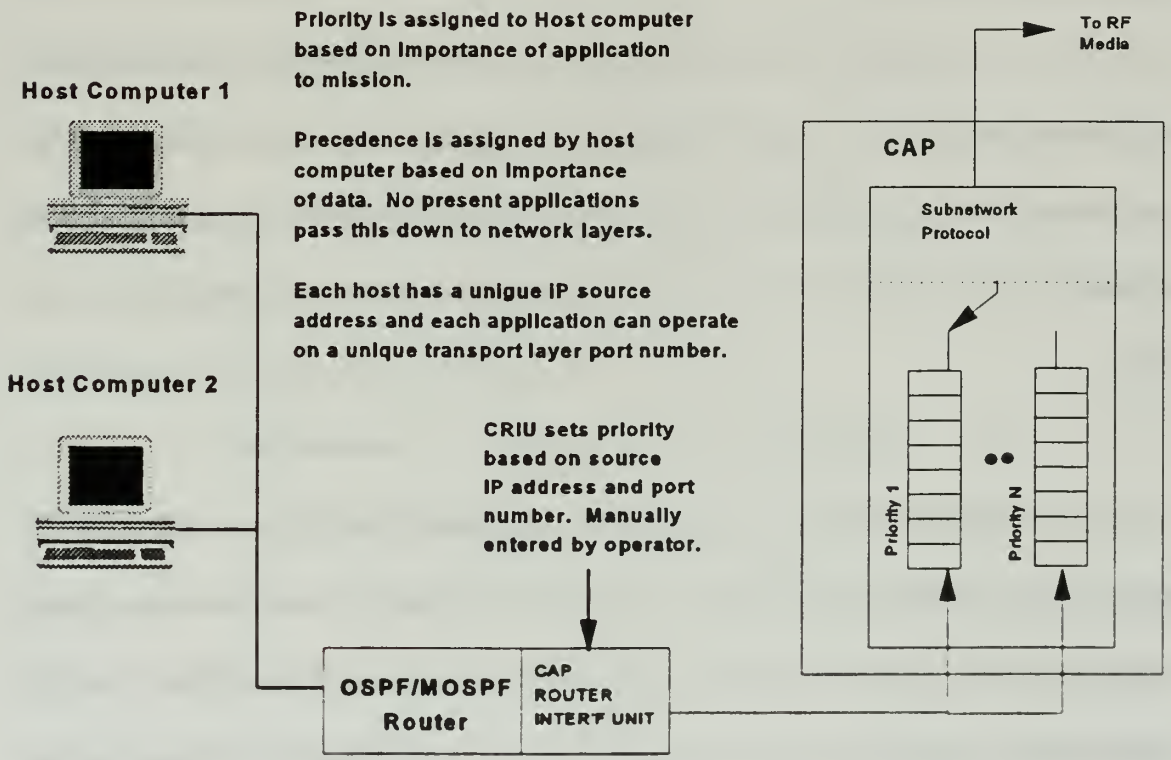


Figure 2.4 Priority Assignment By CRIU (Casey, 1997)

This approach has the same advantages and drawbacks of a firewall using similar data to make its filtering decisions. The advantage is its practicality. The disadvantage is that it's rather crude and currently requires manual configuration of the priority tables.

### *a. Priority Tables*

The CRIU maintains two priority tables. The Source IP table contains the IP addresses of hosts on the associated LAN and the priority which they have been assigned. There are no default settings for this table. If a host is to have an associated priority, it must be entered into the table. This table is filled in manually by the local ADNS Manager during initial system configuration and can be updated at any time. The Source IP table contains space for up to 40 entries.

The second table maintained by the CRIU is the Port Priority table. It contains the dedicated port numbers used by certain applications and the priority that has been assigned to that particular application. Just as with the Source IP table above, there are no default values, priorities must be entered manually, and it contains space for up to 40 entries.

### *b. Determining Message Priority*

The CRIU receives datagrams from the router. The CRIU determines the port number and originating IP address for each datagram and assigns priority based on entries in the Source IP and Port priority tables. Here, a conflict may arise. If the Source IP priority table assigns a certain priority to a particular datagram and the Port priority table indicates a different priority for the same datagram, priority assignment will be made on the basis of

Source IP address. This allows priority based primarily on host, and secondarily on application, should the host have no assigned priority. If neither the host nor the application have an assigned priority, the CRIU assigns a default value of priority 4. Once assigned, the priority is placed in the IP datagram header and the entire IP datagram is passed to the CAP.

#### *c. Message Transmission*

Following Assignment of priority, the IP datagram is forwarded to the appropriate CAP, where it is entered into one of 16 queues based on priority. Datagrams are assembled into transmission units, each of which can contain up to 64 IP datagrams. The size of the transmission unit depends on the capacity of the link. Lower capacity links will use lower transmission unit sizes. The CAP builds a transmission unit by removing datagrams from the queues in order of priority. Datagrams are removed from the highest priority queue first, until it is empty or the transmission unit is full. Datagrams are then removed in sequence, continuing down the priority queues until the transmission unit is complete or all queues are empty. The transmission unit is then sent from the CAP to the corresponding RF transmitter and the process is repeated.

### 4. Load Balancing

Load balancing is the division of transmission load equally among different subnets. When the router selects a transmission path it does so based on the metric assigned to that RF system. OSPF metrics are based on link capacity, with links having similar capacity being assigned identical metric values. If multiple CAPs have the same metrics values then the router will balance the load evenly by alternating between those CAPs. For load

balancing to work effectively, the balancing must be done among systems of equivalent capacity. Consequently, when assigning metric values to RF systems it is important that only networks of like capacity be assigned the same values. For example, suppose a ship is operating two active subnets, HF (which operates at about 2.4Kbps) and SHF (which operates at about 64Kbps). Assigning the same metric values to each would overload the HF circuit. The router would divide the load equally between the two, not in proportion to link capacity. During periods of high traffic density the SHF link is able to handle the load more effectively than the HF link, which would become backlogged with data.

### 5. Congestion Control

As described above, each CAP maintains separate queues for each priority (0-15). Should one of these queues become full, the CAP does not provide any overflow queue. This results in additional datagrams with this same priority being dropped. In order to prevent this situation from occurring, the CRIU monitors the CAP queues and takes action to ease the congestion.

Each queue in a CAP is allocated a certain queue size to store IP datagrams prior to transmission. The CAP manages this queue space. The CRIU contains a queue threshold for each queue, slightly smaller than the queue size, to use as a benchmark to determine if congestion of the queue exists. The gap between the queue threshold and the maximum queue size provides a buffer to allow action to be taken before the queue becomes full and datagrams start being discarded. These queue thresholds are pre-determined and entered into the CRIU by the local ADNS Manager. The congestion identification function operates in

the following sequence. The CAP generates a queue report at intervals specified by the queue report threshold. This report captures the actual queue levels and sends them to the CRIU. These levels are compared to the queue threshold for each queue. If any queue level is greater than the queue threshold, a congestion condition is deemed to exist in that queue. When congestion exists in a queue, the CRIU implements control measures such as Load Sharing or Source Quench in an attempt to alleviate the congestion. The macro behavior of this arrangement is very similar to congested routers in a conventional Internet, indicating that TCP, including the Karn and Nagel algorithms, are applicable in this situation.

*a. Load Sharing*

One of the key features of ADNS is its ability to share the traffic load over available subnets. In current Navy circuits a situation frequently occurs in which one communication channel is overloaded while another is completely idle. The load sharing feature of ADNS alleviates this problem by shifting some of the congestion to the idle channel, thereby increasing throughput and shortening communication system delays. This differs from load balancing in that balancing distributes traffic over channels with similar metric values before congestion occurs. Sharing distributes traffic over similar cost channels because a congestion condition exists.

There are two restrictions on the use of load sharing. First, the traffic being shifted to an alternate channel must be unicast traffic only. Multicast applications introduce a level of complexity that causes diminished returns, making it not worth the effort to attempt to load share using multicast applications. Second, load sharing is only feasible

between subnets whose bandwidths are in the same range, meaning they share a similar time delay. Thus, possible opportunities for a load sharing situation are between UHF and EHF, or between SHF and Challenge Athena.

The load sharing process begins when the CRIU determines that a congestion condition exists on a subnet in one of its associated CAPs. The CRIU then scans all other compatible (those with similar delay times) subnets to determine if a path from origin to destination exists. If another subnet does exist with a path from origin to destination and no congestion condition exists on this subnet, load sharing commences.

*b. Source Quench*

When congestion is determined to exist in the CAP queue for priority n, the CRIU issues a Source Quench ICMP command. This command stops the generation of message packets for all applications and hosts with priority n or less. Assuming compliant TCPs, this Source Quench command has been pre-set to remain in force for five seconds. At the end of five seconds, transmission from the affected hosts and applications resumes automatically unless or until another Source Quench command is issued. It should be noted that all applications and hosts require some sort of flow control to ensure that during Source Quench conditions, packets are not discarded but rather stored for transmission when the Source Quench has timed-out.

### 6. Transmission Control Protocol (TCP) Duplicate Packet Transmission Problems

One of the major early setbacks to implementing the ADNS architecture was solving the problem of TCP duplicate transmissions when initially establishing a TCP connection. ADNS causes the LAN gateway router to act as if it is hard-wired to other routers on other LANs. Thus the router expects to encounter minimal delays (less than 0.5 seconds) in receiving acknowledgments to its TCP packets being sent. In reality, these TCP packets are being transmitted over RF links to distant LANs. The minimum acknowledgment time for a 1500 byte packet over a 2400 BPS connection is in the neighborhood of 5 seconds. When TCP hasn't received packet acknowledgment after 0.5 seconds, it re-transmits the packet. If acknowledgment is still not received after an additional 1 second, TCP retransmits the packet again, and again after 2 seconds, 4 seconds, 8 seconds, and so on. Under optimal conditions, a 1500 byte packet will be sent 4 times over a 2400 BPS connection. The end result is the use of 6000 bytes to transmit 1500, an efficiency of 25%.

#### *a. TCP Duplicate Packet Rejection*

A practical solution, and the one implemented in ADNS, is to design the CRIU to discard duplicate TCP packets before they are transmitted over the RF link. This process is illustrated in Figure 2.5.

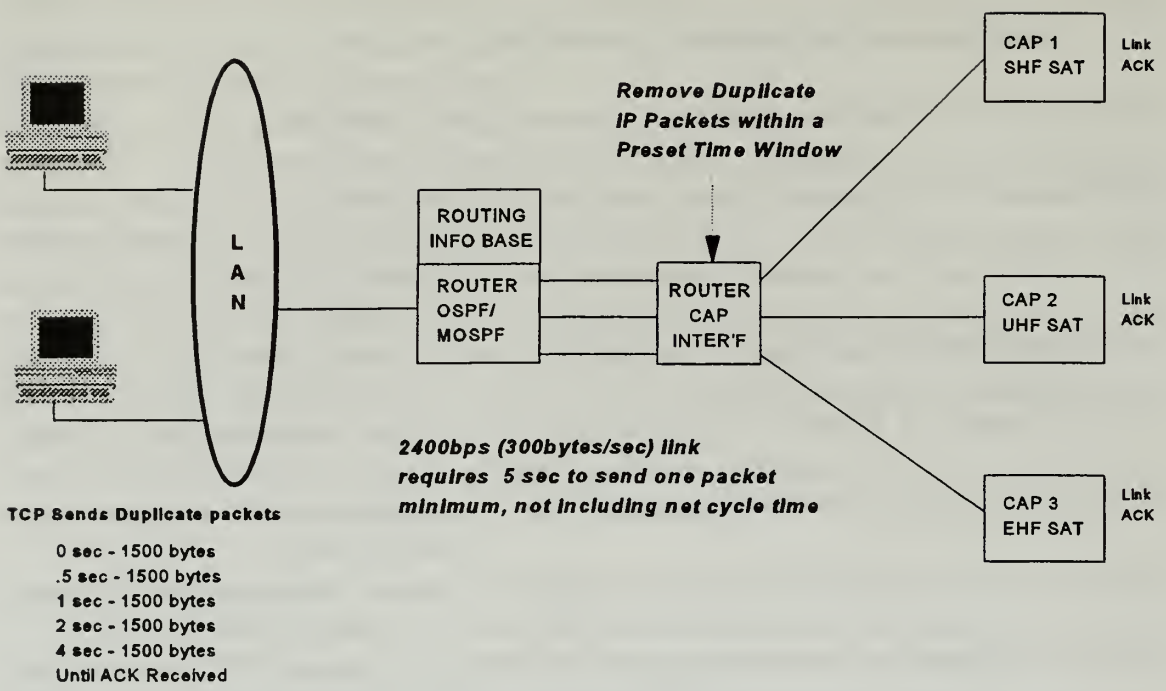


Figure 2.5 TCP Duplicate Packet Rejection (Casey, 1997)

Duplicate packet rejection is accomplished by the use of a table for each subnet that contains the TCP sequence number and time-stamp indicating when the packet was received by the CRIU for transmitting. A TCP original packet and each duplicate packet sent will have the same TCP sequence number. When a TCP packet is received by the CRIU for transmitting, its TCP sequence number is examined. If this number already exists in the table, the packet is rejected. If this number does not exist in the table, it is added to the table along with its time-stamp, and the packet is passed along for transmission. Each subnet is assigned a TCP duplicate rejection time. If a TCP sequence number has been in the table for longer than the TCP duplicate rejection time, it is deleted from the table. The TCP duplicate rejection time has a default value of 10 seconds. This provides for transmission of the original TCP packet followed by a 10 second delay for acknowledgment. If none is received, the packet is allowed to be retransmitted followed by another 10 second delay. This time delay can be modified by the Local ADNS Manager, based on the latency of the link, for optimum performance.

## C. ADNS ADVANTAGES

### 1. Removing Humans From the Loop

In current naval communication systems, messages are generated on personal computers or workstations. These messages are transmitted via LAN, (or by use of magnetic media such as floppy disks where no LAN exists), to the communication center. The messages are then processed by technicians and transmitted. This process introduces time delays ranging from minutes to hours. ADNS eliminates the need for human processing of

messages by establishing a direct connection from any node on the LAN, through the transmitter, to the receiver at the intended destination. The result is complete automation of the transmission process, with total elimination of any handling delays caused by human interaction.

### 2. Load Sharing

Most naval vessels maintain at least two operational communication channels at all times, often substantially more. These multiple, incompatible communication paths came about as new communication systems were added over time in response to emerging requirements. The situation frequently exists where one or more channels are completely silent, while others are backlogged with traffic. The Load Sharing Feature of ADNS was specifically designed to alleviate these backlogs by making more efficient use of all operational communication channels. As an added bonus, the Load Sharing mechanism of ADNS is completely transparent to the user and completely automatic in operation, requiring no human intervention to be initiated.

### 3. Optimal Use of Bandwidth

Network costs are assigned such that higher capacity circuits are assigned lower cost values. ADNS maximizes throughput by finding the lowest cost path for a message to reach its destination. The combination of removing humans from the loop, load sharing and using the lowest cost paths discussed above results in a four-fold increase in throughput during peak traffic times (Kvigne, 1997). This is a direct increase in the bottom line throughput of the communications system without purchasing additional transmitters.

### 4. Communications Agility

ADNS provides the capability for two units that do not share a common communication channel to maintain communications. As long as each unit is operating at least one communication channel and at least one node on the network is operating both channels simultaneously, communications can occur. This process is completely transparent to the users, and occurs with no human intervention. This is analogous to Internet packet delivery. Few end systems share a common communications channel (that is, they exist on the same network segment), yet data can be shared between these end systems through intermediate nodes and communication paths. This is a significant increase in communications capability which is not available current systems.

### 5. Transparency of Installation and Use

The installation of ADNS is totally transparent to the end users. It merely appears that a new router has been added to the LAN with links to many other LANs. There is no major LAN or transmitter reconfiguration that is required. The entire ADNS installation is small and lightweight, allowing it to be installed in any unused space without impacting shipboard weight and balance. Additionally, there are no major infrastructure modifications required, (cooling, ventilation, etc.), and power requirements are modest.

### 6. Ease of upgrade

Following initial installation, upgrading of ADNS is quite simple. Addition of new communication channels can be accomplished through the installation of the appropriate CAP cards. Adding capabilities to ADNS itself, such as installing successive builds as they

become available, is as simple as downloading the new software. Router reconfiguration is a relatively simple matter as well.

### 7. Single Point for Communications Management

ADNS provides a single point for monitoring all communications, both incoming and outgoing. Prior to ADNS, monitoring all communications was much more difficult due to the lack of interconnection between stovepipe systems. Each of these systems had to be monitored separately. This monitoring capability is available locally via the local net manager's workstation, or remotely from the Network Operations Center.

### 8. Ability to Transmit All Types of Data

Essentially, ADNS transmits Internet Protocol (IP) datagrams from one router to another. It is the applications residing on the LANs connected to these routers that decode the datagrams and put the information contained in them to use. Therefore, ADNS can transmit text, graphics, voice, or video applications over existing channels, without the need for developing expensive new stovepipe systems to support each new application.

## D. ADNS INTEGRATED NETWORK MANAGEMENT

### 1. Overview

Network management of ADNS is based on SNMPv1 standards. There are no proprietary Navy protocols to develop, implement, and maintain. This allows the use of standard network management tools and practices. Most of the objects to be managed (hosts, routers, etc) have agents attached, and Management Information Bases (MIBs) will be written for any unique objects. The Navy will adopt a standard, commercial Network

Management System (NMS) to provide the foundation for network management. However, there are Navy-specific concerns, such as command and control relationships, which impact network management. For these special requirements, the Navy will create special applications and concepts for the NMS.

Network management of naval nodes is similar to managing shore-based nodes. The fundamental concepts are the same. However, the mobile nature of the nodes makes managing shipboard nodes more difficult. The fact that they are warships makes management more important. Just as there is a military hierarchy, there is also a hierarchy for network management in ADNS, where each level has different responsibilities. Network management is a vital element of ADNS because the consequences of system errors or failures can directly affect combat effectiveness.

Integrated network management describes how the Navy will manage networks on a distributed basis all the way down to individual objects. This includes, but is not limited to: general monitoring, statistic collection, status monitoring, traffic monitoring, trend analysis, network loading, network optimization, configuration control, system configuration, maintenance, problem identification, problem reporting, trouble documentation, system administration, and emissions control.

### 2. Levels of Network Management

Network management of ADNS is accomplished at three different levels: the Local Control Center (LCC), Autonomous System Control Center (ASCC), and the Navy Operations Center (NOC). The LCC will be responsible for networks at the local level, e.g.

within an area (usually a ship). The ASCC will be in charge of networks on a regional level, having several subordinate Autonomous Systems. The NOC will be responsible for all ASCCs in a certain geographic area. This arrangement is consistent with Navy organization and doctrine regarding distribution of authority.

### *a. Local Control Center (LCC)*

The LCC is the network management center at every unit level. There is a local responsibility to monitor and maintain the status of all subnets at that unit. There are three components of an LCC: a Network Manager, Distributed Manager and a Communication Automation Manager.

(1) Network Manager. The Network Manager consists of NMS software that is obtained commercially. The purpose of the Network Manager is to provide the status of the network and individual objects. One example of an NMS is the popular HP Open View Network Node Manager product (OV-NNM) which has been specified in the Navy Tactical Advanced Computer (TAC) contracts since 1991. It provides a topological map representation of a unit's network and shows the status of each object with the use of colors and shapes. However, software cannot provide complete net management, human interaction is required to interface with the ASCC and the NOC for troubleshooting or maintenance. The specific functions of a Network Manager will be:

* Human machine interface
* Performance management

* Fault management

* Accounting management

* Security management

* Configuration management

The Network Manager will be used as the foundation for the Navy's Integrated Network Management System, specific applications can then be added to provide additional management functions.

(2) Distributed Manager. Distributed Management is an application that determines which information is to be reported locally and which is to be reported to the ASCC and NOC. The Distributed Manager has two mechanisms for discovering any conditions that meet the criteria of its policy rules:

* Notification from the Network manager

* Query from Distributed Manager to Network Manager

The specific functions of the Distributed Manager will be:

* Interpretation and implementation of policy

* Filtering of management information

Although commercial products can provide these functions, the distributed manager in the Navy context specifically describes the policy rules for the communication relationships between the LCC and ASCC.

(3) Communication Automation Manager (CAM). The Communication Automation Manager is in charge of the physical communication hardware

and their related requirements. Aboard ship, these functions are typically related to the radio room. Duties include a communication plan implementation, circuit building, and circuit management. Three areas make up the Communication Automation Manager: the Communication Manager, Site Manager, and Equipment Manager. The specific functions of the Communication Automation Manager will be:

* Security management
* Log Control
* Alarm reporting
* Summarization
* Attributes for representing relationships
* Objects and attributes for access control
* Usage Metering
* Test Management
* Event Report Management
* State Management
* Security alarm reporting
* Object management
* Bandwidth management
* Communication plan management
* Equipment control
* Site configuration management

The Navy specific application for these functions is the use of a remote management tool called the Communications Plan (COMMPLAN). The COMMPLAN will used to direct certain network management functions as described above. This is still mainly accomplished manually by a technician after receiving the COMMPLAN via hardcopy message. However ADNS will allow many of these requirements to be accomplished remotely and automatically via the COMMPLAN transmitted to the Communication Automation Manager. This concept allows simultaneous, automated update of all nodes in the AS ensuring constant communications between all nodes.

### *b. Autonomous System Control Center (ASCC)*

An ASCC monitors the operation of several LCCs. The Navy's configuration will use its regional shore communications master stations (NCTAMS) as ASCCs. The ASCC will receive summary reports from subordinate LCCs. The exact nature of reporting from an LCC to an ASCC is still to be determined but will contain mission relevant information. Such reporting requirements can include:

* Readiness of communication to support the mission.
* Status of communication services.
* Status of hardware and software.
* Information about usage and reliability.

ASCCs can also give direction to LCCs regarding communications posture. This could include such items as prioritization of resources or equipment configuration changes.

### *c. Network Operations Center (NOC)*

The NOC is the next level above an ASCC for reporting network management information. The NOC monitors all nodes in a certain geographic location. For example, the Navy has established separate NOCs in the Pacific and Atlantic regions. Although capable of monitoring detailed network management information, a NOC would primarily be concerned with the overall status of ASCCs and LCCs.

## 3. The Need for Remote Network Management

ADNS is a good example of the need for remote management. Implementation of remote management over ADNS will allow managers to configure and manage mobile platforms from a central management location. This, in turn, allows the assignment of minimal personnel at the local level, thus saving on personnel costs. With such standards as RMON and SNMPv2, central managers can access remote networks in a secure manner and troubleshoot or reconfigure the network. For example, if one transmission path fails, a central manager can gain access to the system remotely via a second transmission path and troubleshoot the system. The use of more than one transmission path allows the ability to continually manage LCCs and even ASCCs remotely through a single open path. Although ADNS has not adopted such standards as RMON or SNMPv2 yet, the technologies currently exist and can be readily integrated into ADNS.

# III. ANALYSIS OF NAVAL COMMUNICATIONS

In order to provide a useful analysis of quality of service over ADNS, the recent history of naval message traffic and ADNS must first be reviewed. This review will serve as the basis for a discussion on the current validated communications requirements of the fleet users as well as predictions of future requirements. This discussion will include current and planned communications capabilities mapped against validated requirements. This chapter will conclude with a section which stresses the importance of timely, verifiable delivery of essential message traffic.

## A. HISTORY

ADNS was developed, in part, as a response to observations of message traffic flow during the Persian Gulf War. During the war, each user had a designated slice of bandwidth assigned for use, either exclusively or shared with other users in some form of time division multiplexing. It was observed that certain communications channels were suffering huge backlogs of message traffic, while other channels experienced only occasional heavy use interspersed with long periods of little or no use. This was generally regarded as a necessary evil. In order to allow high priority, mission essential, or time critical message traffic to be transmitted within acceptable time frames, this message traffic was given dedicated bandwidth.

> The dedication of a Radio-Frequency (RF) channel has created the impression among the users that such dedication is necessary for timely message delivery.

> This is particularly true for users of the TACINTEL system, who are very much concerned with timely delivery of very short messages and are not particularly concerned with the efficient utilization of the RF channel. (Melich,1980)

This bandwidth was commonly allocated to support the largest predicted burst of traffic, even though this would occur infrequently. The end result was a large percentage of wasted bandwidth. ADNS allows recoupment of this wasted bandwidth, with a documented economy of bandwidth of approximately 4 to 1 as illustrated in Figure 3.1 (CRWG, 1997).

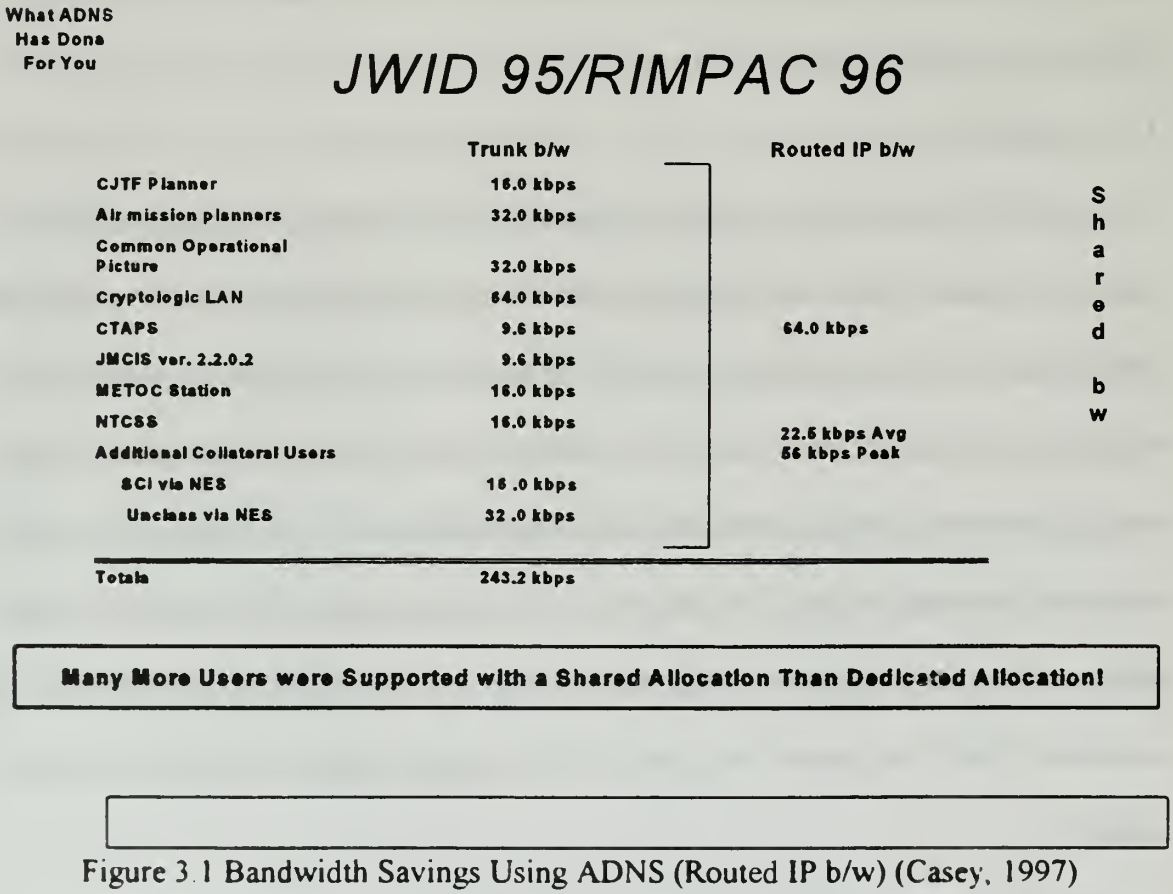


Figure 3.1 Bandwidth Savings Using ADNS (Routed IP b/w) (Casey, 1997)

## B. PRESENT AND FUTURE COMMUNICATION REQUIREMENTS

In recent history, the requirements for naval communication have always outstripped existing capabilities. This is partially the reason for the conglomeration of stovepipe communication systems in existence today. These systems were rushed through development in order to meet a pressing need with no thought given to interoperability. Current requirements and predictions for future needs continue to follow this trend, as illustrated in Figure 3.2 (CRWG, 1997).

The estimates illustrated in Figure 3.2 are pre-ADNS requirements estimates. It would appear that the 4 to 1 economy of bandwidth provided by ADNS would satisfy predicted requirements. However, this 4 to 1 economy of bandwidth represents only one data point. Even if these preliminary results are accurate over all cases, there are other factors to consider before victory can be declared in meeting predicted communications requirements.

Parkinson's Law has been stated in many ways, but can generally be paraphrased as, "Work will expand to fill the time allotted." A corollary to Parkinson's Law applicable to Naval communications can be stated as "Applications will expand to fill the bandwidth available." This is especially true given the current explosion in Internet technology and emphasis on IT-21 in today's Navy. As this technology is pushed further and further down the chain of command, traffic will increase at a greater and greater rate.

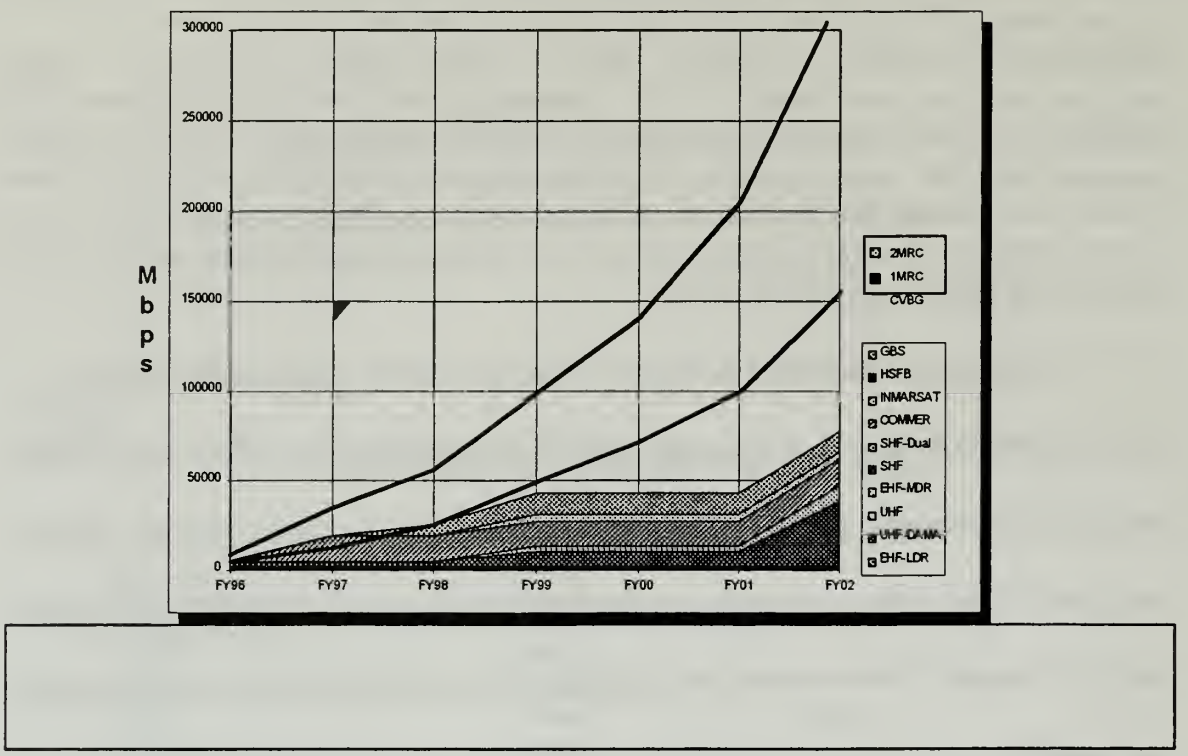


Figure 3.2 Bandwidth Requirements vs. Available Capacity (CRWG, 1997)

## C. IMPORTANCE OF DETERMINISTIC SERVICE

Deterministic service places bounds on time of delivery and acceptable error rate to satisfy the requirements of time or lost-packet sensitive applications. Deterministic service in the commercial world refers to the premise that a user can be reasonably assured that his message packets being sent over an Internet will be received within a given time frame and in an accurate and reliable manner. Deterministic service has come into being to support applications which, by their very nature, or the information they convey, are time sensitive.

These applications cannot tolerate their message packets sitting in router queues waiting for other, lower priority, or non-time-sensitive applications to be transmitted ahead of them.

The need for deterministic service came about as terrestrial internets became crowded with more and more users transferring larger and larger amounts of data. This, in turn, slowed the rate at which an individual transaction could take place. In situations where time-sensitive data (e.g. real-time audio or video) was being transmitted over crowded links, the data transfer rate was dropping below the minimum threshold for un-interrupted transfer. This resulted in datagrams arriving too late for the application to use. These datagrams are then discarded, causing gaps in service and loss of information. Deterministic service can prevent this loss from occurring.

Deterministic service is even more important in a Wide Area Network (WAN) implemented over Radio Frequency (RF) links. A WAN Operating System is required to perform certain functions and applications on a time bounded basis. If a particular node or application does not receive service within a specified time period, it may time-out and be declared inoperative causing a loss of functionality or connectivity.

The need for deterministic service appears at many levels in the conducting of military operations. First, in the world of Joint Operations, force coordination is imperative. Deterministic service provides the ability to ensure that widely dispersed units receive imperative information regardless of network loading. Second, weapons coordination among Joint forces requires deterministic service to ensure that each threat or target is adequately countered by the appropriate level of force. In this case, deterministic service will permit the

real-time handoff of targets to the most appropriate weapon system, thereby optimizing use of assets and ensuring full coverage. Also, deterministic service will enable precise accuracy in the transmission of launch orders to widely dispersed platforms, ensuring weapons launched from these platforms impact the targets simultaneously. Finally, deterministic service is also highly applicable in the area of weapons control. In the case of beam riding missiles, updates must be received periodically. If a missile does not receive input within a pre-determined time interval, it will abort the mission and self destruct. In this last case, speed of transmission is not the issue, the real issue is bounded service.

With such widespread applications for deterministic service, the Internet Engineering Task Force (IETF) has developed deterministic service standards. These standards, known as Quality of Service (QoS) standards, are explained in detail in Chapter IV.

# IV. QUALITY OF SERVICE STANDARDS

## A. DEFINITION

Quality of Service (QoS) refers to the nature of the packet delivery service provided, as described by parameters such as achieved bandwidth, packet delay and packet loss rates (Shenker, Wroclawski, 1996). The most basic QoS level is classified as Best Effort. This is analogous to "First-come, First-served." Message packets are entered in a first-in-first-out (FIFO) queue at the source node. Packets are removed from the queue to be transmitted in the order in which they were received. The time spent in the transmission queue is known as queuing delay. This process is repeated at each node along the transmission path, thereby adding a queuing delay for each node to the total transmission time for that particular message packet. It was quickly realized that queuing delay represents the bulk of total transmission time, and that any effort to improve QoS may do so by reducing queuing delay. Further, any effort to guarantee a certain level of QoS must do so by placing an upper bound on queuing delay. Efforts to improve on Best Effort QoS have produced several additional levels of service, among which are Controlled Load QoS, and Guaranteed QoS. The requirements which a setup mechanism must follow to achieve Controlled Load QoS and Guaranteed QoS are given in Sections B and C below. Setup mechanisms, which include reservation protocols and scheduling algorithms, are not specified in these requirements, but will be addressed in Chapter V.

The information presented in Chapters III and IV is largely taken from Internet Drafts on the subjects of QoS and Resource Reservation Protocol (RSVP). Internet Drafts are working papers of the Internet Engineering Task Force (IETF). They represent work still in progress and in many cases the services and protocols being described have not been thoroughly tested and evaluated for general use. However, Internet Drafts are extremely useful advance notice as to what is on the horizon in the Information Technology field. Taken in this context, this thesis presents the descriptions of QoS and RSVP provided in these Internet Drafts, and offers these services and protocols as a possible solution to priority and congestion problems with ADNS.

## B. CONTROLLED LOAD QUALITY OF SERVICE

The basic principle of Controlled Load QoS is to provide service which closely resembles that received by message packets under the Best Effort QoS on an unloaded network. In the context of this thesis, unloaded refers to a network that is not overloaded or congested, rather than a network with absolutely no other traffic. In a properly functioning network, Controlled Load QoS specifies the following:

-A very high percentage of transmitted packets will be successfully delivered by the network to the receiving end-nodes. (The percentage of packets not successfully delivered must closely approximate the basic packet error rate of the transmission medium.)

-The transmit delay experienced by a very high percentage of the delivered packets will not greatly exceed the minimum transmit delay experienced by any successfully delivered packet. (This minimum packet delay includes speed-of-light delay plus the fixed

processing time in routers and other communications devices along the path.) (Wroclawski, 1997)

### 1. Situations for Use of Controlled Load QoS

Controlled Load QoS was developed to provide network transmission services to a class of applications which are very sensitive to overloaded network conditions. Members of this class include "Adaptive Real-time Applications" which operate efficiently on unloaded networks, but degrade rapidly under overloaded conditions (Wroclawski, 1997). Controlled Load QoS was designed with no optional functions or capabilities. This minimalist design allows it to be implemented in a variety of methods, including admission control scheduling algorithms which allow the most efficient use of bandwidth and network resources.

### 2. Node Requirements

Each node in the transmission path between sending and receiving nodes (inclusive) must meet minimum requirements in the areas of node resources, queuing delay, congestion loss, bandwidth, and burst handling.

#### *a. Node Resources*

The components of each node must be capable of supporting the Controlled load QoS Routers and switches must provide adequate port buffer space, and the packet forwarding engine must possess adequate computational capacity, for example.

### *b. Queuing Delay*

Controlled Load QoS requires little or no average packet queuing delay over all time scales significantly larger than the burst time. Burst time is the time required for the flow's maximum size data burst to be transmitted at the flow's requested transmission rate. (Wroclawski, 1997)

### *c. Congestion Loss*

Controlled Load QoS requires little or no congestion loss over all time scales significantly larger than the burst time. In this context, congestion loss includes packet losses due to shortage of any required processing resource, such as buffer space or link bandwidth. The wording "little or no" allows for occasional congestion losses, but any sustained loss represents a failure of the admission control algorithm (Wroclawski, 1997).

### *d. Bandwidth*

A network node must allocate bandwidth for each Controlled Load service request it receives. However, a node that processes a number of Controlled Load flows simultaneously and has done so for an extended period of time may be able to use analysis of past behavior to over allocate its available bandwidth to a small extent. This is similar to the way airlines overbook flights based on past occurrences of no-shows. However, unlike airlines, there is no "later flight" to handle overflows, so over allocation algorithms must be extremely conservative.

The over allocation described above may not be as risky as it sounds due to the fact that nodes must allocate more bandwidth to each flow than the Controlled Load

service request specifies. To see why this is required, consider a request that specifies a certain transfer rate and maximum packet size. Suppose this flow is allocated exactly enough bandwidth to support transfer of the maximum packet size at the requested rate. Now suppose a random transfer delay causes an over-rate burst to be received by a node, followed by a resumption of traffic at the limit of the allocated bandwidth. This burst will cause a queuing delay which will never clear, which is in direct violation of Controlled Load QoS. Allocation of bandwidth slightly more than that called for by the Controlled Load QoS request is a conservative solution which will aid in dealing with moderate burst conditions.

*e. Burst Handling*

In order to effectively deal with heavy traffic bursts, the Controlled Load QoS requires each node to provide a mechanism to borrow the bandwidth required in order to clear bursts from the network. This may be an explicit borrowing scheme within the traffic scheduler or an implicit scheme based on statistical multiplexing and measurement-based admission control.

In addition to bandwidth flexibility, each node is required to provide some sort of buffer flexibility. In the presence of bursty flows of traffic, buffer space in excess of the maximum packet size is required. This may be supplied by multiplexing of a shared buffer pool. Once again, a conservative algorithm for multiplexing should be implemented to ensure adequate buffer space is available to all flows which require it.

### 3. Invocation Information

Controlled Load QoS is invoked by providing the network nodes with the desired traffic parameters of the data flow. These parameters, contained in the TOKEN_BUCKET_TSPEC, are: bucket rate (r), bucket depth (b), minimum policed unit (m), Maximum packet size (M), and peak rate (p).

#### *a. Bucket Rate (r)*

Bucket rate is a positive rate measured in bytes of IP datagrams per second with a range from 1 byte per second to 40 terabytes per second. Network nodes must return an error for any request that falls outside of this range, as well as for any request within this range that cannot be supported by an individual node on the transmission path. The range of values allowed for bucket rate as well as bucket depth is intentionally large to allow for future network technologies, any given network element is neither required nor expected to support the full range of values.

#### *b. Bucket Depth (b)*

Bucket depth is a positive value measured in bytes with a range from 1 byte to 250 gigabytes. Network nodes must return an error for any request that falls outside of this range, as well as for any request within this range that cannot be supported by an individual node on the transmission path.

#### *c. Minimum Policed Unit (m)*

The minimum policed unit is an integer measured in bytes. Any IP datagrams forwarded for transfer containing less than m bytes are counted as having m bytes against

the token bucket. The minimum policed unit must be less than or equal to the maximum policed unit.

*d. Maximum Policed Unit (M)*

The maximum policed unit is also an integer measured in bytes. It represents the maximum packet size to be transferred as specified in the Controlled Load QoS service request. A node on the transmission path must reject a service request if the specified maximum policed unit is larger than the maximum transfer unit of the link.

*e. Peak Rate (p)*

The peak rate has the same range and form as the bucket rate and exists in the TOKEN_BUCKET_TSPEC for compatibility with other QoS specifications, it is not currently used by the Controlled Load QoS specification.

*f. Preferred Representation*

The preferred representation of the TOKEN_BUCKET_TSPEC, specified in order, is bucket rate (r), bucket size (b), peak rate (p), minimum policed unit (m), and maximum policed unit (M).

**4. Exception Handling**

The Controlled Load QoS is provided to a data flow on the basis that the flow conforms to the TOKEN_BUCKET_TSPEC. The flow must obey the rule that for all time periods t, the amount of data transmitted does not exceed (rt+b). Packets that violate this rule are considered nonconformant. When nonconformant packets are detected, each node must ensure the following requirements are met.

-The node must continue to provide contracted QoS to all Controlled Load flows not experiencing nonconformant traffic.

-The node should prevent nonconformant Controlled Load traffic from impacting the handling of arriving Best Effort traffic.

-Consistent with the two requirements above, the node must attempt to forward nonconformant traffic on a best effort basis if sufficient resources are available (Wroclawski, 1997.)

Network nodes must not assume that arrival of nonconformant traffic for a specific Controlled Load flow will be unusual or indicative of error. In certain circumstances (particularly routers acting as split points of a multicast distribution tree supporting a shared reservation) large numbers of packets will fail the conformance test "as a matter of normal operation" (Wroclawski, 1997).

The Controlled Load QoS specification merely requires that each node conform to the three requirements listed above, but does not specify the exact handling of flows with nonconformant traffic. There are two methods for forwarding nonconformant traffic, degraded service or flow sorting.

### *a. Degraded Service*

Degraded service implies that the service received by all packets in the flow containing a nonconformant packet is downgraded to best effort. This will increase queuing delay and packet loss probability, but keep packet re-ordering at low levels.

This method of service is preferable for applications that treat out of order packets as lost packets, triggering retransmissions.

*b. Flow Sorting*

Flow sorting allows the segregation of packets in a flow into a conformant group, which continues to receive Controlled Load QoS, and a nonconformant group, which will receive degraded service such as Best Effort. This will allow the conformant group of packets to arrive with low loss and delay, while the nonconformant group arrives with potentially higher loss and delay. This method is suited to applications which are time-critical but tolerant of out-of-order or lost packets.

**5. Examples of Use**

The Controlled Load QoS may be used by any application which can make use of Best Effort service, however it is much better suited to applications which can accurately characterize their traffic requirements. Applications which transmit continuous media such as digitized audio or video are prime examples. It is important to note that Controlled Load service is not isochronous, but most applications which produce digitized audio or video provide a timing recovery mechanism which is already used with Best Effort service. Other applications which are suited to Controlled Load QoS are those which are sensitive to overloaded conditions. These applications may request Controlled Load QoS upon start-up, or only when performance begins to degrade due to congestion (Wroclawski, 1997).

## C. GUARANTEED QUALITY OF SERVICE

The basic principle of Guaranteed QoS is to provide guaranteed delay and bandwidth to data flows over an IP network. The service provided by Guaranteed QoS at a guaranteed bandwidth R must closely approximate the service provided by a dedicated wire of bandwidth R between source and receiver, to within tight error bounds. This implies that the service provided to a data flow under Guaranteed QoS is completely independent of the service provided to other data flows. In order to determine the maximum delay that can be guaranteed by the Guaranteed QoS, the components of end-to-end delay must first be explored.

### 1. Components of End-to-End Delay

In order to place an upper bound on total end-to-end delay, upper bounds must first be determined for each component. As mentioned previously in chapter IV, section A, the majority of transmission (end-to-end) delay can be attributed to queuing delay. The remainder can be contributed to the latency of the communication path.

#### *a. Queuing Delay*

An upper bound on queuing delay is calculated by starting with the time required for a token bucket of depth b to be transmitted over a share of bandwidth of size R. This time is represented by b/R. To this, error terms are added to account for maximum node deviation. The first term, represented by C/R, accounts for node delays related to the data transfer rate R. The second term, represented by D, accounts for node delays independent of transfer rate. The final queuing delay induced by a particular node is therefore bounded

by the expression (b/R + C/R + D) (Shenker, et al., 1997). These terms are described in further detail in Subsection 4, Subsubsections a and b.

### *b. Latency of the Communication Path*

The latency of the communication path is a term used to account for all other causes of delay not included in queuing delay. This type of delay is not addressed by any QoS service, but a conservative estimate can be made by observing the delay experienced by any one packet during the QoS request process. Though this type of delay is outside the scope of a QoS service, methods of controlling the sources of this delay are discussed briefly in the following paragraphs.

At the data link layer, frame size affects latency. Larger frame size increases throughput (less frame header per byte of data), but increases latency by causing a longer wait until the end of a frame, at which point a new, higher priority frame can be transmitted. Also at the data link layer, token holding time has a similar effect. Longer token holding time increases throughput, but shorter token holding time decreases latency. These tradeoffs must be made according to the situation at hand.

At the physical layer, some coding algorithms affect latency. The most obvious of these are forward error correction and interleaving. Forward error correction adds redundant bits to the data stream so that loss of one or two bits can be corrected. Interleaving spreads the redundant bits out so that a single "hole" in the data stream represents a group of single-bit errors instead of a one multiple-bit error. Single bit errors are easily correctable whereas multiple bit errors are not. The depth of interleaving is directly proportional to latency.

Though these sources of latency are somewhat controllable, for the purposes of QoS evaluation they will be considered to be constant and a conservative estimate of total latency of the communication path will account for all these sources.

### 2. Node Requirements

Each node in the transfer path from source to receiver must ensure that the service provided closely approximated the "fluid model" of service. The fluid model at service rate R is essentially the service that would be provided by a dedicated wire of bandwidth R between the sender and the receiver. The flow's level of service is characterized at each node by the share of bandwidth and buffer space it is entitled to. The node must ensure that enough bandwidth and buffer space is allocated to the data flow so that service provided by the node matches the service provided by the fluid model to within a sharp error bound. Nodes are not permitted to fragment datagrams in order to meet Guaranteed QoS requirements (Shenker, et al., 1997).

### 3. Invocation

Guaranteed QoS is invoked by an application by specifying to the network nodes the nature of the traffic to be transmitted and the level of service desired. The TOKEN_BUCKET_TSPEC specifies the nature of the traffic to be transmitted. The RSPEC specifies the level of service desired.

*a. TOKEN_BUCKET_TSPEC*

The parameters contained in the TOKEN_BUCKET_TSPEC, are: bucket rate (r), bucket depth (b), minimum policed unit (m), Maximum packet size (M), and peak rate (p), as defined in Chapter IV, Section B, Subsection 3.

*b. RSPEC*

The RSPEC contains two parameters: requested rate of transfer (R), and a slack term (S). The rate of transfer is measured in bytes of IP datagrams and is subject to the same range and representation as the bucket and peak rates. The requested rate of transfer (R) must be greater than or equal to the bucket rate (r). The requested rate of transfer may exceed the TSPEC rate because higher rates will reduce queuing delay. (Shenker, et al., 1997) The slack term is a nonnegative number measured in microseconds. It represents the difference between the delay encountered when transmitting at the TSPEC rate and transmitting at the RSPEC rate. This slack term can be used by the node to reduce its resource reservation for this flow. When a node chooses to use some of the slack in the RSPEC, it must update the (R) and (S) fields in the RSPEC.

There is no buffer space allocation parameter in the RSPEC. Each node is expected to calculate buffer requirements from the incoming TSPEC and RSPEC using its own characteristics as a guideline to ensure Guaranteed QoS can be provided.

**4. Exported Information**

Each node in the transmission path must be capable of providing or calculating , and passing on, the following information: rate dependent error term (C), rate independent error

term (D), total rate dependent error term (Ctot), total rate independent error term (Dtot), sum of rate dependent error since last reshaping point (Csum), sum of rate independent error since last reshaping point (Dsum).

*a. Rate Dependent Error Term (C)*

The rate dependent error term accounts for all queuing delays which vary according to the rate at which a flow is transmitted. An example is the need to account for the time taken serializing a datagram broken up into ATM cells, with the cells sent at a frequency of 1/r (Shenker, et al., 1997).

*b. Rate Independent Error Term (D)*

The rate independent error term accounts for all non-rate-based delays. An example is a slotted network, in which guaranteed flows are assigned particular slots in a cycle of slots. Some part of the per-flow delay may be determined by which slots are allocated to the flow. In this case, D would represent the amount of time a flow's data, once ready to be sent, might have to wait for a slot. In this case, the upper bound on D would be the time it takes to cycle through all the slots.

*c. Total Rate Dependent Error Term (Ctot)*

The total rate dependent error term is the sum of the rate dependent error terms (C) for each node in the transmission path.

*d. Total Rate Independent Error Term (Dtot)*

The total rate independent error term is the sum of the rate independent error terms (D) for each node in the transmission path.

*e. Sum of Rate Dependent Error Since Last Reshaping Point Term (Csum)*

The sum of rate dependent error term since the last reshaping point is the sum of the rate dependent error terms for each node in the transmission path since the last reshaping point. Reshaping points are discussed in Subsection 5 below.

*f. Sum of Rate Independent Error Since Last Reshaping Point Term (Dsum)*

The sum of rate independent error term since the last reshaping point is the sum of the rate independent error terms for each node in the transmission path since the last reshaping point. Reshaping points are discussed in Subsection 5 below.

**5. Policing**

Policing refers to the monitoring of packets to ensure they conform to the specifications provided in the TSPEC. Nonconformant packets should be marked and forwarded as Best Effort traffic. However, it is permissible, under Guaranteed QoS, to discard nonconformant packets if this action is tolerable to the application generating the service request. The two forms of policing that exist in Guaranteed QoS are simple policing and reshaping.

*a. Simple Policing*

Simple policing is comparing of arriving packets with the TSPEC to ensure they conform with all TSPEC parameters. All packets which do not conform to the TSPEC parameters will be identified as nonconformant and dealt with according to the implementation mechanism. Simple policing is done at the edges of the network.

### *b. Reshaping*

Reshaping is an attempt to restore a traffic flow's parameters to conform to the TSPEC. It is done by combining a buffer with a token bucket and peak rate regulator and buffering data until it can be sent in conformance with the token bucket and peak rate parameters. The amount of buffer space required to reshape any nonconforming traffic back into its original token bucket shape is given by:

$$(b + Csum + (Dsum*r))$$

The network node must provide the necessary buffers to ensure that all conforming traffic is not lost at the reshaper (Shenker, et al., 1997). Failure of the reshaping process, in which the reshaping buffer overflows, indicates that the traffic is in violation of the TSPEC, and will be identified as nonconformant.

Reshaping is done at all heterogeneous source branch points and at all source merge points. A heterogeneous source branch point is a spot where the multicast distribution tree from a source branches to multiple distinct paths, not all of which have the same TSPEC. Reshaping need only be done on a branch if the TSPEC reserved for that branch is smaller than the TSPEC reserved for the immediate upstream link. A source merge point is the point where the distribution paths from two separate sources (sharing the same reservation) merge. This situation will likely occur where a terrestrial internet interfaces with a Radio-WAN, precisely the situation created by ADNS. The mechanism which invokes the Guaranteed QoS is responsible for identifying the points which require reshaping (Shenker, et al., 1997).

It would appear that reshaping adds a large amount of delay, but given a valid TSPEC that accurately describes the traffic, this is not true. It is possible, however, that with multiple Guaranteed QoS paths between common sources and destinations, and with many merge and branch points, the TSPEC may be smaller than the actual traffic. ADNS is likely to induce extreme examples of this situation. This will cause large reshaping queues to form, introducing delays and causing traffic to be identified as nonconforming. Therefore, mechanisms using the Guaranteed QoS for multicast applications must calculate TSPECs and monitor flows to avoid this situation. This monitoring and re-shaping must be given particular consideration in the case of ADNS.

### 6. Operation

Guaranteed QoS accepts a TSPEC and RSPEC as input and provides a guarantee of both maximum delay and bandwidth available for a data flow. This QoS provides an upper limit of delay as a result rather than accepting a delay bound as an input to be achieved. If this upper limit of delay is unacceptable to an application, that application can modify its token bucket depth and data rate and re-submit its TSPEC and RSPEC to achieve a lower delay. In order to achieve this level of service, every node in the path from the source to the receiver must support Guaranteed QoS. Because this service is so tightly bounded and resource intensive, some sort of admission control is recommended to permit only those applications with hard, real-time requirements to have access. For example, some audio/video play-back applications are intolerant of any packet arriving after the play-back time. These would be good candidates for Guaranteed QoS.

The requirements for Controlled Load and Guaranteed QoS listed above are merely basic requirements that each node and application must fulfill to support implementation of QoS. In order to actually implement a specific QoS, a Resource Reservation Algorithm must provide the mechanisms to gather, transmit, and evaluate the information provided by nodes and applications. This Algorithm will then make resource reservations based on the information provided. One such Resource Reservation Algorithm, the Resource Reservation Protocol (RSVP) is described in detail in Chapter V.

# V. RESOURCE RESERVATION PROTOCOL (RSVP)

## A. DESCRIPTION OF SERVICE

RSVP is a protocol designed to implement QoS mechanisms defined in the integrated services framework. Currently, RSVP supports only Guaranteed QoS and Controlled Load QoS, but additional levels of QoS defined by the IETF are expected to be supported by future versions of RSVP. Specifically, RSVP provides a method to transport and utilize the node and application information specified by the various QoSs to set up resource reservations for the applications requesting them. In general terms, RSVP evaluates the data to be transmitted, evaluates the capabilities of the transmission path, reserves resources along the path, and provides head of the line service for privileged data flows at each node along the path to ensure timely delivery. The addition of RSVP to ADNS will provide for deterministic service, a feature for which there is currently no provision in present and future builds of ADNS. In this sense, RSVP is an add-on for ADNS, providing enhanced capabilities upon request while having no negative impact on the functionality of ADNS.

## B. SUMMARY OF OPERATION

The basic sequence of events in establishing a particular QoS starts when an application instance wishing to participate in an RSVP session registers with the RSVP agent. The information provided in this registration is bundled into an RSVP PATH message and transmitted from the source node, along the transmission path, to the receiving node. Each node along the way provides information on its capabilities and capacities which is

recorded and forwarded in the RSVP PATH message. Upon arrival at the receiving node, the data in the RSVP PATH message is passed through the RSVP agent to the receiving application. The application, possibly with the assistance of a common library of RSVP setup functions, interprets the data and generates resource reservation parameters. These parameters, including desired QoS, are passed to the RSVP agent and transmitted back up the original transmission path as a FLOWSPEC message. At each node along the way, resources are reserved for this particular data flow. When the FLOWSPEC message reaches the originating RSVP agent, the session's data sender(s) are given the required information on the type of QoS that has been arranged (Guaranteed or Controlled Load) and characteristics of the data path. The data sending applications are then free to begin transmission under the agreed upon QoS. The exact composition of the RSVP_PATH and the FLOWSPEC messages are discussed in detail below.

## C. RSVP PATH MESSAGE

### 1. Generation

The RSVP PATH Message consists of the RSVP SENDER TSPEC, and the RSVP ADSPEC.

#### *a. RSVP SENDER TSPEC*

The RSVP SENDER TSPEC carries the traffic specification (in the form of a TOKEN BUCKET TSPEC defined in Chapter IV, Section B, Subsection 3) generated by each data source within an RSVP session. This object conveys basic data flow information such as maximum size of data packets and transfer rate, to each node in the transfer path as

well as to the data receivers. For a complete description of the RSVP SENDER TSPEC, see Appendix A.

*b. RSVP ADSPEC*

The RSVP ADSPEC is generated at the data source(s) and initially contains data about the QoS control capabilities and requirements of the data source(s). This initial RSVP ADSPEC forms the starting point for the accumulation of QoS capability information for each node in the transmission path. This information includes both parameters describing the properties of the data path, (including the availability of specific QoS control services), and parameters required by specific QoS control services to operate correctly (Wroclawski, 1997). The RSVP ADSPEC will contain a section for each different QoS supported by the data sender(s). For a complete description of the RSVP ADSPEC see Appendix B.

**2. Transmission**

The RSVP PATH Message is passed from node to node along the transmission path. At each node, the resident RSVP agent passes it to the traffic control module. The traffic control module identifies each QoS in the RSVP ADSPEC and calls that service in the node to update its portion of the RSVP ADSPEC. It may be possible that a service mentioned in the RSVP ADSPEC is not supported by a particular node. If this is the case, the traffic control module sets a flag bit to indicate this to the receiver(s). Once all supported QoS services have updated their portions of the RSVP ADSPEC, the RSVP PATH Message is returned to the RSVP agent for forwarding to the next node where the process is repeated.

### 3. Receipt

Upon receipt of the RSVP PATH Message at a data receiver, the RSVP SENDER TSPEC and the RSVP ADSPEC are passed across the RSVP agent to the application. The application uses the data in these objects to determine resource reservation parameters. This may be done with the help of a library of common resource-reservation functions (Wroclawski, 1997). Key parameters which are calculated include path maximum transfer unit for both Controlled load and Guaranteed QoS, and "C" and "D" parameters (Chapter IV, Section C, Subsection 4, SubSubsections a and b) to calculate a mathematical bound on delay for Guaranteed QoS.

## D. RSVP FLOWSPEC

Once calculated, key parameters for a QoS request are assembled into an RSVP FLOWSPEC Message.

### 1. Generation

The RSVP FLOWSPEC contains a Receiver TSPEC which describes the level of traffic for which resources should be reserved, and a Receiver RSPEC ( if requesting for Guaranteed QoS) which describes the level of service desired by the receiver.

#### *a. Receiver TSPEC*

The Receiver TSPEC is of the same format as the TOKEN BUCKET TSPEC defined in Chapter IV, Section B, Subsection 3. The values for bucket rate (r), bucket size (b), peak rate (p), and minimum policed unit (m) are taken directly from the RSVP SENDER TSPEC. The value for the Maximum policed unit (M) is calculated as the minimum of the

RSVP SENDER TSPEC Maximum policed unit, and the Maximum transfer units of each node in the transmission path which are obtained from the RSVP ADSPEC. In this way, no RSVP session will specify a Maximum policed unit which is larger than the maximum transfer capability of any node on the transmission path.

*b. Receiver RSPEC*

The Receiver RSPEC is used for the Guaranteed QoS. It contains the same information as the RSPEC defined in Chapter IV, Section C, Subsection 3, Subsubsection b. The requested rate (R) represents the transfer rate desired by the data receiving application.

**2. Transmission**

The FLOWSPEC Message is transferred from node to node back up the transmission path. At each node, the RSVP agent passes the FLOWSPEC Message to the traffic control module, which calls the requested QoS to make the resource reservations required for this particular data flow. At this point, state merging, message forwarding, and error handling occur according to the rules of the RSVP protocol (Wroclawski, 1997).

**3. Receipt**

Upon receipt of the FLOWSPEC Message at the data sending application(s), the reservation process is complete. Provided that there were no error flags set, (e.g. a node on the transmission path does not support RSVP, or sufficient resources to complete the reservation were unavailable) the application has now been granted the requested QoS.

So long as the message packets transmitted by the application adhere to the SENDER TSPEC, they will receive the agreed upon QoS.

## E. CHOICE OF SERVICE CONSIDERATIONS

While the reservation process itself is not bandwidth intensive, the act of reserving bandwidth and buffer space so that certain data flows receive privileged service places constraints on traffic flow. This is analogous to allowing certain applications head-of-the-line privileges. If too many applications are granted head of the line privileges, there will be no bandwidth and buffer space available to service non-privileged traffic. Also, if all available bandwidth and buffer space is allocated to privileged applications, no additional (and perhaps more important) applications can be granted privileged service. It is for these reasons that some form of access control must be implemented. Only those applications which by their very nature (real time video applications), or content (vital, time sensitive reports) require some form of QoS should be granted access to RSVP. Ideally, these applications should be further limited to a pre-determined percentage of available resources. This will allow the transmission of traffic which only rates Best Effort service as well as allowing some flexibility for new RSVP sessions to be opened up should an unforseen need arise which requires RSVP to provide a certain QoS.

It is worth noting that RSVP sets up a fixed path from source to destination. This is exactly the situation that robust, connectionless IP has avoided. Therefore, failure of a node along the transmission path of an RSVP connection can be expected to cause problems in maintaining any type of QoS. As of this writing, RSVP has only been evaluated over stable,

terrestrial networks. Any implementation of RSVP over ADNS would necessarily have to provide some sort of soft-fail capability in which failure of a node along the transmission path caused, at the very worst, a reversion to Best Effort QoS.

# VI. ANALYSIS, CONCLUSIONS, AND RECOMMENDATIONS

## A. ANALYSIS

In order to determine whether implementing QoS standards with RSVP is appropriate for naval communications, several factors must be considered. First, it must be determined if naval communications applications fit the profile of applications suitable to take advantage of QoS services. Also, the feasibility of implementing QoS mechanisms and the RSVP protocol in the ADNS architecture must be studied. Finally, the impact of QoS on the user community must be taken into account. Each of these topics is covered in detail in the Subsections below.

### 1. Application Suitability

In determining if an application is suitable for and will benefit from being included in an RSVP session, three questions must be answered. Is the information vital? Is it hard real-time? Are the packets of predictable size and being produced at a predictable rate?

Vitality of information is obviously situation dependent, however, certain applications can be deemed vital by their very nature. An application such as threat reporting from a radar picket ship may fall into this category. This information is vital to battlegroup security and must be received no matter what other traffic is inbound.

This information is also hard real-time in that threat reporting will contain threat information as well as possibly containing real-time radar video of the threat. If these real-time video packets are delayed long enough to fill available buffer space, information will

be lost. This may occur if another high priority application, such as the transfer of vital intelligence photos, happens to occur at the same time as the threat report.

Finally, the packets produced by a threat reporting application will be of predictable size and be transmitted at a predictable rate, both of which are requirements for QoS services. Other applications which meet the requirements of being vital, time sensitive and predictable fall into the general category of real-time voice or video applications. The fleet is currently experimenting with video teleconferencing which also fits this category.

### 2. Feasibility of Implementation

Implementation of QoS services over ADNS starts with the procurement of devices which support the IETF QoS standards. Because QoS standards are in the public domain, these routers already exist in the commercial marketplace. Procurement of these routers is possible due to the fact that the ADNS project is in its infancy, with funding decisions for fleet installations currently being made. Contract wording which permits the purchase of upgradable routers as technology improves should allow for this option.

The second phase of QoS implementation is the actual loading of RSVP agents into the ADNS software. This is also possible due to the current ADNS build plan. Each successive build will incorporate more options and features. Testing and evaluation of RSVP can easily be added to the build plan in Builds 2 or 3 scheduled for the near future.

Finally, any discussion of the feasibility of implementing QoS must take into account the fact that QoS only delivers the promised service if all nodes in the communications path support QoS services. On the surface, the installation of QoS supporting hardware/software

in every node on every network appears to be an immense task. However, taken node by node, (e.g. Ships, Battlegroups, Fleets), the capability can be installed in stages. The larger problem, however, will be support for QoS over terrestrial networks which interface with Radio-WANs. Providing QoS support over NIPRNET and SIPRNET really is an immense task, but the alternative may be information gridlock, especially given the current explosion in information technology.

### 3. Impact of RSVP on the User Community

The implementation of QoS services using RSVP will have a profound effect on the user community. It will add a level to the communications plan which details the applications and situations eligible for QoS services. This will occur in addition to assigning priorities of zero through fifteen for designated applications and users. The actual applications and situations designated to receive QoS services will be a subject of intense debate. RSVP has the effect of creating a class of privileged applications. Care must be taken to ensure that these applications are chosen on the basis of suitability for RSVP sessions discussed in Subsection 1 above, not on the basis of rank or position of the users of these applications.

## B. CONCLUSIONS

The following are conclusions of this research drawn from and supported by the preceding chapters.

### 1. Network Congestion Will Necessitate QoS

Bandwidth requirements for naval communications will continue to increase at a faster and faster rate. The availability of new bandwidth is costly and limited by technology and shipboard considerations. ADNS helps utilize bandwidth in an efficient manner, but will still inevitably crowd the network with more and more traffic. During times of peak usage, such as in a combat scenario, high priority applications will compete with each other for bandwidth. It is precisely during these situations that time-sensitive high-priority applications must be given QoS. This will ensure data does not time-out while in a buffer waiting for other high priority, non-time-sensitive applications to complete their transmissions.

### 2. A COTS, Network Centric Solution is Indicated

Great strides have been made in using ADNS to transform a stovepipe communications configuration into a network centric communications system utilizing COTS hardware. This trend must be continued in further developing ADNS to better meet the needs of the fleet user. The adoption of COTS software, such as RSVP, to enhance the capabilities of ADNS is definitely the right way to proceed. This allows for a shorter (or non-existent) research and development phase, commercial support, lower cost, and access to future upgrades.

### 3. RSVP Offers Flexible Solution to Problem

RSVP provides a flexible solution to the problem of ensuring that high-priority, time-sensitive applications receive the QoS they require. It is flexible in that RSVP currently

supports both Controlled Load and Guaranteed QoS, with support for other IETF classes of QoS being developed for future versions. This allows the user to determine what level of QoS is desired and therefore what constraints are placed on available resources. Choosing the appropriate level of QoS will ensure that traffic arrives in a timely manner for that particular application regardless of the network load, while minimizing the constraints placed on available resources. This is important in that fewer constraints allows non-QoS traffic to receive better service.

## C. RECOMMENDATIONS

The following recommendations will facilitate the adoption of RSVP as the mechanism to implement QoS standards on ADNS.

### 1. Incorporate QoS Supporting Hardware Into ADNS Builds

Routers that support the IETF QoS standards are available today and should be incorporated into all future ADNS installations. Because the QoS standards are in the public domain, they are being incorporated into products by many different companies and should eventually be a market standard just as smart routers that support SNMP management have become.

### 2. Backfit QoS Supporting Hardware and Software Into Existing ADNS Installations

The number of existing ADNS installations is still small, making the backfitting of QoS supporting routers a feasible option in order to gain the assurance of applications receiving the QoS they require in order to operate effectively under all conditions of network loading.

### 3. Enlist Key Agencies and Programs to Support QoS and RSVP

For QoS standards and RSVP to be effective, they must be implemented end-to-end over communications networks. Therefore, DISA (which oversees SIPRNET) must be enlisted to support the adoption of QoS standards and RSVP. Further, these standards must be included in the IT-21 architecture, at pierside connections, and at shore-based commands throughout the Navy.

## D. QUESTIONS FOR FURTHER STUDY

-Explore modification of CRIU to support IETF QoS standards and RSVP agent.

-Explore feasibility of adding support for IETF QoS standards to CAPs/modems.

-Explore feasibility of adding support for IETF QoS standards to NIPRNET and SIPRNET.

-Explore management issues pertaining to application eligibility for QoS, specifically:

Identification of applications suitable for QoS.

Identify type of QoS suitable for each application.

Identify situations in which QoS is required for applications identified above.

Create guidelines for use of QoS in normal/emergency conditions.

-Determine condition of RSVP QoS upon failure of a node in the transmission path.

## APPENDIX A. RSVP SENDER TSPEC OBJECT

| | 31 24 | 23 16 | 15 8 | 7 0 |
|---|---|---|---|---|
| 1 | 0 (a) | reserved | 7 (b) | |
| 2 | 1 (c) | reserved | 6 (d) | |
| 3 | 127 (e) | 0 (f) | 5 (g) | |
| 4 | Token Bucket Rate [r] (32-bit IEEE floating point number) | | | |
| 5 | Token Bucket Size [b] (32-bit IEEE floating point number) | | | |
| 6 | Peak Data Rate [p] (32-bit IEEE floating point number) | | | |
| 7 | Minimum policed unit [m] (32-bit integer) | | | |
| 8 | Maximum packet size [M] (32-bit integer) | | | |

(a) - Message format version number (0)

(b) - Overall length (7 words not including header)

(c) - Service header, service number 1 (default/global information)

(d) - Length of service 1 data, (6 words not including header)

(e) - Parameter ID, parameter 127 (TOKEN BUCKET TSPEC)

(f) - Parameter 127 flags (none set)

(g) - Parameter 127 length, (5 words not including header)

## APPENDIX B. RSVP ADSPEC OBJECT

| | 31 24 | 23 16 | 15 8 | 7 0 |
|---|---|---|---|---|
| 1 | 0 (a) | Unused | 19 (b) | |
| 2 | 1 (c) | x Reserved (d) | 8 (e) | |
| 3 | 4 (f) | (g) | 1 (h) | |
| 4 | zero extension of .. | | IS hop cnt (16 bit unsigned) | |
| 5 | 6 (I) | (j) | 1 (k) | |
| 6 | Path b/w estimate (32-bit IEEE floating point number) | | | |
| 7 | 8 (l) | (m) | 1 (n) | |
| 8 | Minimum path latency (32-bit integer) | | | |
| 9 | 10 (o) | (p) | 1 (q) | |
| 10 | zero extension of .. | | Composed MTU (16 bit unsigned) | |
| 11 | 2 (r) | reserved (s) | 8 (t) | |
| 12 | 133 (u) | (v) | 1 (w) | |
| 13 | End-to-end composed value for C [Ctot] (32-bit integer) | | | |
| 14 | 134 (x) | (y) | 1 (z) | |
| 15 | End-to-end composed value for D [Dtot] (32-bit integer) | | | |
| 16 | 135 (aa) | (bb) | 1 (cc) | |
| 17 | Since last reshaping point composed C [Csum] (32-bit integer) | | | |
| 18 | 136 (dd) | (ee) | 1 (ff) | |
| 19 | Since last reshaping point composed D [Dsum] (32-bit integer) | | | |
| 20 | 5 (gg) | 0 (hh) | 0 (ii) | |

Word 1: Message Header
(a) - Message header and version number
(b) - Message length - 19 words not including header

Words 2-7: Default general characterization parameters
(c) - Per-Service header, service number 1 (Default General Parameters)
(d) - Global break bit ( NON_IS_HOP general parameter 2) (marked x)
(e) - Length of General Parameters data block (8 words)
(f) - Parameter ID, parameter 4 (NUMBER_OF_IS_HOPS general parameter)
(g) - Parameter 4 flag byte
(h) - Parameter 4 length, 1 word not including header
(I) - Parameter ID, parameter 6 (AVAILABLE_PATH_BANDWIDTH general parameter)
(j) - Parameter 6 flag byte
(k) - Parameter 6 length, 1 word not including header
(l) - Parameter ID, parameter 8 (MINIMUM_PATH_LATENCY general parameter)
(m) - Parameter 8 flag byte
(n) - Parameter 8 length, 1 word not including header
(o) - Parameter ID, parameter 10 (PATH_MTU general parameter)
(p) - Parameter 10 flag byte
(q) - Parameter 10 length, 1 word not including header

Words 11-19: Guaranteed QoS parameters
(r) - Per-service header, service number 2 (Guaranteed)
(s) - Break bit
(t) - Length of per-service data, 8 words not including header
(u) - Parameter ID, parameter 133 (Composed Ctot)
(v) - Composed Ctot flag byte
(w) - Composed Ctot length, 1 word not including header
(x) - Parameter ID, parameter 134 (Composed Dtot)
(y) - Composed Dtot flag byte
(z) - Composed Dtot length, 1 word not including header
(aa) - Parameter ID, parameter 135 (Composed Csum)
(bb) - Composed Csum flag byte
(cc) - Composed Csum length, 1 word not including header
(dd) - Parameter ID, parameter 136 (Composed Dsum)
(ee) - Composed Dsum flag byte
(ff) - Composed Dsum length, 1 word not including header

Word 20: Controlled Load QoS parameters
(gg) - Per-service header, service number 5 (Controlled Load)
(hh) - Break bit
(ii) - Length of controlled-load data, 0 words not including header

## LIST OF REFERENCES


Casey, Roger. Navy Research and Development (Code D8205), ADNS Implementation Working Paper, NRaD, July 15, 1997.

Casey, Roger, and Stell, Mark. Navy Research and Development (Code D8205), Autonomous System Implementation for Navy Afloat Forces, NRaD, June 26, 1997.

CRWG, (Copernicus Requirements Working Group), Automated Digital Network System (ADNS) Information Brief, Space and Naval Warfare Systems Command, PMW-176, May 12, 1997.

Landwehr, Carl. Naval Research Lab, Washington, D.C., Performance Studies of the Distributed CPODA Protocol in the Mobile Access Terminal Network, September, 1979.

Tran, Trung. Navy Research and Development, Class Notes-CAP Router Interface Unit Training Course, NRaD, May 22, 1997.

Shenker, S., Partridge, C., and Guerin, R. IETF Integrated Services Working Group, Specification of Guaranteed Quality of Service, Internet-Draft, February 3, 1997.

Wroclawski, J. IETF Integrated Services Working Group, Specifications of the Controlled-load Network Element Service, Internet-Draft, May, 1997.

Wroclawski, J. IETF Integrated Services Working Group, The Use of RSVP With IETF Integrated Services, Internet-Draft, July, 1997.

## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center ..... 2
   8725 John J. Kingman Rd., Ste 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library ..... 2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, CA 93943-5101

3. Joe Macker ..... 1
   Code 5544
   Center for High Assurance Computer Systems
   Navy Research Lab
   Washington, D.C. 20375

4. Roger Casey ..... 1
   NRaD (Code D8205)
   271 Santa Catalina Blvd.
   San Diego, CA 92152

5. Brain Clingerman ..... 1
   SPAWAR, PMW-176
   53560 Hull St.
   San Diego, CA 92152

6. Mike Sovereign ..... 1
   HQ CINCPAC, J56
   Box 64015
   Camp H.M. Smith, HI 96861-4105

7. Rex Buddenberg ..... 1
   Naval Postgraduate School
   Code SM/BU

8. Brian Rehard ..... 1
   5461 Cresthaven Ln., Apt. A4
   Toledo, OH 43614